Applying machine learning to the dynamic selection of replenishment policies in fast-changing supply chain environments

Applying Machine Learning to the Dynamic Selection of Replenishment Policies in Fast-Changing Supply Chain Environments

Firms currently operate in highly competitive scenarios, where the environmental conditions evolve over time. Many factors intervene simultaneously and their hard-to-interpret interactions throughout the supply chain greatly complicate decision making. The complexity clearly manifests itself in the field of inventory management, in which determining the optimal replenishment rule often becomes an intractable problem. This paper applies machine learning to help managers understand these complex scenarios and better manage the inventory flow. Building on a dynamic framework, we employ an inductive learning algorithm for setting the most appropriate replenishment policy over time by reacting to the environmental changes. This approach proves to be effective in a three-echelon supply chain where the scenario is defined by seven variables (cost structure, demand variability, three lead times, and two partners' inventory policy). Considering four alternatives, the algorithm determines the best replenishment rule around 88% of the time. This leads to a noticeable reduction of operating costs against static alternatives. Interestingly, we observe that the nodes are much more sensitive to inventory decisions in the lower echelons than in the upper echelons of the supply chain.

Keywords: Bullwhip Effect, inductive learning, inventory management, machine learning, replenishment policy, supply chain management.

1. Introduction

Globalization has utterly changed the business landscape, where competition has not only increased substantially but also become more complex and dynamic (Puche et al. 2016). This competition has indeed moved from the firm level to the network level, placing a premium upon *supply chain management* as a key source of competitive advantages (Melnyk et al. 2009). However, these advantages are difficult to capture. Managers must deal with distant partners —geographically, culturally, and administratively—, control convoluted supply networks with long and variable lead times, and be able to agilely react to the frequent changes in the environment (Mentzer et al. 2001). Comprehending the supply chain interdependencies between processes, decisions, and structures is far from being trivial, which makes decision making a challenging task.

The complexity becomes evident in the field of *inventory management*, one of the cornerstones of the supply chain discipline. APICS (2011, 48) defines inventory as "an expensive asset" that "needs to be carefully managed", whose primary purpose is "to meet demand in support of production or customer service". In this sense, managers need to evaluate two primary aspects when making replenishment decisions to control the inventory flow (Disney and Lambrecht 2008). First, they must consider a key trade-off between inventory investment and service level, with the aim of satisfying consumer demand in a cost-effective manner (Steinker, Pesch, and Hoberg 2016). Second, they need to examine the production implications of replenishment rules, which determine the variability of production schedules and hence may trigger different sources of costs, e.g. extra capacity, overtime, and idle time (Disney et al. 2006). Overall, Lancioni (2000) claimed that inventory-related costs cover nearly 50% of the supply chain costs.

Under these circumstances, determining a suitable replenishment policy is key to the performance of supply chains. To this end, managers need to consider the impact of the complex interactions between a wide range of variables, which may result in an intractable problem (Bischak et al. 2014). This task becomes even more difficult in what we label as *fast-changing supply chain environments*, in which the conditions defining this environment (e.g. consumer demand, raw materials cost, or stakeholders' decisions) suffer from frequent changes over time (Chopra and Sodhi 2004). In these cases, it may be necessary to react to these changes by modifying the replenishment policy, which questions the performance of traditional static approaches to inventory management.

From this perspective, this work develops a dynamic framework for managing inventories in the supply chain. The framework employs machine learning, specifically inductive learning, for understanding the complex relationships between the controllable and uncontrollable factors that impact on business performance. It has been designed to periodically select the best inventory policy, among a set of baseline rules, according to the environmental conditions at every moment. To illustrate our approach, we compare its performance against traditional static alternatives in a simulated case study. We aim to show that machine learning can help managers make decisions that are hard to deal with from other approaches, which eventually would result in an increased performance. In this sense, machine learning techniques may be interpreted as a promising next step in the field of inventory management.

This paper is structured as follows. Section 2 provides an overview of the inventory problem —we focus on the measurement of performance and present some established replenishment policies. Section 3 introduces the inductive algorithm that we use and delves into previous applications of machine learning to supply chain management. Section 4 describes the dynamic framework we propose for managing inventories. Section 5 presents the case study where we test our proposal, and details the generation of examples for the learning algorithm. Section 6 shows the numerical results and evaluate them against the static alternative. Finally, Section 7 concludes and reflects on the implications of this research.

2. Inventory management: Metrics and policies

2.1. Measuring operational performance through the Bullwhip Effect

In the management of inventories throughout the supply chain, practitioners face a powerful enemy: the *Bullwhip Effect* (Lee, Padmanabhan, and Whang 1997). This phenomenon is common in all kinds of industries (see e.g. Isaksson and Seifert 2006) and may reduce the profitability of firms significantly (Metters 1997). It refers to the tendency of the variability of the signals, mainly orders and consequently inventories, to increase as they pass through the various nodes of the supply chain; see the recent review by Wang and Disney (2016) for further detail. From the previous definition, two ratios are commonly used to quantify the Bullwhip Effect: the order variance ratio (*OVR*) and the inventory variance ratio (*IVR*). The former compares the variance of the orders issued (σ_O^2) and received by the node, i.e. its demand (σ_D^2), by eq. (1); while the latter quantifies the variance of the net stock[1] (σ_{NS}^2) against the demand variability, by eq. (2).

$$OVR = \frac{\sigma_O^2}{\sigma_D^2} \tag{1}$$

$$IVR = \frac{\sigma_{NS}^2}{\sigma_D^2} \tag{2}$$

As previously discussed, decision makers need to consider both the production and inventory implications of inventory management policies. Interestingly, the previous metrics cover both aspects; thus defining a powerful framework for evaluating the operational performance of supply chains (Cannella et al. 2013). First, OVR measures order variability, which is highly undesirable as it tends to create unstable production schedules that significantly decrease supply chain efficiency. Indeed, Disney, Gaalman, and Hosoda (2012) showed that the minimum production cost is proportional to the square root of OVR in linear guaranteed-capacity models[2]. Second, IVR considers net stock variability, which determines the firm's ability to meet effectively a predetermined service level. Hence, reducing IVR is essential to appropriately balance the risk of breaking stock and the charge of holding too much stock. In this sense, Kahn (1987) showed that the minimum inventory cost is linearly related to the square root of the net stock (and thus IVR) when holding and backlog costs are proportional to the volume.

The function J fuses both indicators into one metric, through a weighted sum of their square roots; see eq. (3). Here w_o and w_i ($w_o, w_i \geq 0,\ w_o + w_i = 1$) depend on the cost associated to each source of variability and express the relative importance of each indicator. For example, $w_o = 0.8$ ($w_i = 0.2$) would reveal that order variability is more damaging; while $w_o = 0.2$ ($w_i = 0.8$) would illustrate the opposite scenario. Following from the previous discussion, it can be assumed that J provides a fair understanding of the cost performance of a determined inventory police. For this reason, we employ this metric in this work. For further details on J, please refer to Ponte, Wang et al. (2017).

$$J = w_o \cdot \sqrt{OVR} + w_i \cdot \sqrt{IVR} \tag{3}$$

2.2. Managing the inventory flow: the order-up-to policies

To control the inventory, there are several types of replenishment strategies (see e.g. Zipkin 2000). This paper is concerned with the *order-up-to* (OUT) family, which review inventories and place orders at fixed intervals. These periodic-review systems are generally easier to implement and less expensive to operate than continuous-review systems (Axsäter 2003). They also produce benefits from other perspectives; for instance, they enable combined orders to save transportation costs (APICS 2011). Hence, it is a common practice in many industries to forecast and replenish inventory frequently

(Sillanpää and Liesiö 2018) and OUT policies are widely used in real supply chains (Dejonckheere et al. 2003).

OUT policies place orders periodically, e.g. at the end of each period t, to bring the inventory up to a determined level. The traditional OUT model (e.g. Disney and Lambrecht 2008) considers the forecasted demand ($\widehat{D}_t$) and places the order (O_t) to fully recover two gaps, by eq. (4). First, between the safety stock (SS_t) and the actual net stock (NS_t); and second, between the desired and the actual work-in-progress (DW_t, AW_t). Note that work-in-progress covers the product that has been ordered but not yet received.

$$O_t = \widehat{D}_t + [SS_t - NS_t] + [DW_t - AW_t] \tag{4}$$

When the safety stock is appropriately adjusted, the OUT model finds the optimal balance between holding and backlog costs (Karlin 1960). In this sense, this policy is able to minimize the IVR metric. Nevertheless, it generally offers poor performance from the perspective of production-related costs. In this regard, Dejonckheere et al. (2003) proved that the OVR generated by this policy is always greater than 1 for three common forecasting methods. To sum up, Gaalman (2006, 1284) states that the OUT policy "will mainly minimize inventory costs or equivalently inventory variance", but "the control of the order variance is limited".

For this reason, several authors proposed to incorporate a proportional controller β ($0 \leq \beta \leq 1$) into the ordering rule to regulate the amount of gaps to be recovered; e.g. Lin et al. (2017) reviews several applications of inventory controllers over the last decades. This results in the so-called *proportional order-up-to* (POUT) policy, see eq. (5). Depending on the value of the controller, this policy allows modeling a wide range of real-world replenishment strategies (Li and Disney 2017). The smaller β, the less sensitive the order to the inventory gaps. This simple mechanism allows to directly control, and reduce, order variability (Disney and Lambrecht 2008). Gaalman (2006) concluded that the POUT model is always able to generate OVR lower than 1.

$$O_t = \widehat{D}_t + \beta \cdot [SS_t - NS_t] + \beta \cdot [DW_t - AW_t] \quad (5)$$

When $\beta = 1$, the POUT model simplifies to the traditional OUT model. If β is reduced, OVR tends to decrease at the expense of an increase in IVR; e.g. see Figure 1 in Ponte, Sierra et al. (2017). Therefore, reducing β allows managers to decrease ordering-related costs, generally at the expense of increasing inventory-related costs. In light of this, the tuning of the controller has become a fruitful area of study with the aim of finding the right balance between both metrics; see e.g. Cannella and Ciancimino (2010). However, the mathematical complexity of determining the optimal value in real-world settings is very high, generally being an intractable problem through analytical techniques (see e.g. Disney et al. 2006). In this paper, we consider the impact of a wide range of uncontrollable and controllable factors, and their interplays, on determining a suitable value for the controller.

3. Machine learning and its applications in supply chain management

3.1. Machine learning and inductive learning: An overview

Machine learning, belonging to the field of artificial intelligence, explores the development of algorithms capable of learning from data. These techniques can be applied to solve different kinds of problems using knowledge obtained from similar past problems (Michalski, Carbonell, and Mitchell 1983). According to the review by Priore et al. (2014), the main machine learning techniques are: (1) inductive learning; (2) artificial neural networks; (3) case-based reasoning; (4) support vector machines; and (5) reinforcement learning. They diverge in how knowledge is stored. In inductive learning, knowledge results in a set of decision rules that build a decision tree. Thence, this conceptual approach allows users to easily understand the decision-making process (Filipic and Junkar 2000). Next, we describe how it operates, which is outlined in Figure 1.

Insert Figure 1 about here

The learning algorithm obtains the knowledge by examining a training dataset. This includes the past problems and their solutions (examples) and can be represented as an attribute-value table. The input attributes refer to the features of the problem, while a special attribute named "class" includes the optimal solution. Inductive learning techniques recursively split this initial dataset into subsets depending on the value of one attribute. This results in the generation of the decision tree, which is employed to solve new problems by assigning a class to the set of values of the attributes defining them. Note that information about the solved problems may thus be used to analyze future problems. In this sense, this approach incorporates principles of *information updating*, which is gaining interest as an important process for supply chain learning (Shen, Choi, and Minner 2018).

From the pioneering works by Hoveland and Hunt in the 1950s, there is a wide range of inductive learning algorithms. The CART (Friedman 1977), ID3 (Quinlan 1979), PLS (Rendell 1983), ASSISTANT 86 (Cestnik, Kononenko, and Bratko 1987), and C4.5 (Quinlan 1993) deserve to be mentioned here. The last one is generally considered the most popular inductive learning algorithm (Wu et al. 2008; Witten et al. 2016), as it can achieve a very good trade-off between error rate and speed of learning (Lim, Loh, and Shih 2000). For this reason, we employ this algorithm in this research work.

The C4.5 algorithm uses the concept of information entropy to sequentially select the nodes of the tree. This refers to the amount of information produced by a source of data and can be formally expressed by eq. (6) for a set D of cases, where C denotes the number of classes. Note $p(D, j)$ refers to the proportion of cases in D that belong to the j-th class, and $\log_2(\cdot)$ is the logarithmic function with base 2.

$$H(D) = -\sum_{j=1}^{C} p(D, j) \cdot \log_2\big(p(D, j)\big) \qquad (6)$$

This algorithm employs the following divide-and-conquer procedure (Wu et al. 2008). First, it checks if either all the cases in the dataset S belong to the same class or S is small. If so, it simply creates a leaf node for the tree with the most frequent class. Otherwise, it calculates the information gain (the change in information entropy compared to the previous state) from splitting on each attribute A^X and creates a node based on the attribute that maximizes the information gain. This can be maximized in absolute terms (g_1) or in relative terms to the information provided by the test sources (g_2, which corrects the gain by considering information about the class)[3]. Then, it recurs on the obtained subsets through the same procedure. Last, the tree is pruned from the leaves to the root to avoid overfitting. We refer the interested reader to Wu et al. (2008) for more details on the pruning algorithm.

3.2. Applying machine learning to the supply chain: A review

Supply chain management has become more information intensive as a response to the complexity and dynamism of the current business scene. Accordingly, practitioners and academics have explored ways to better manage the information and leverage this to make more robust decisions; e.g. see the review by Ko, Tiwari, and Mehnen (2010). In line with the previous discussion, machine learning can be of special interest in this regard. Next, we review the relevant literature that applies these techniques to the control of inventories in the supply chain. These studies represent the background of our research work.

Several works propose machine learning-based frameworks for managing the inventory at all nodes of the supply chain in a coordinated manner, such as Giannoccaro and Pontrandolfo (2002), Chaharsooghi, Heydari, and Zegordi (2008), and Mortazavi, Khamseh, and Azimi (2015). Their solutions employ different algorithms for reinforcement learning, e.g. Q-learning (Watkins and Dayan 1992), to determine near-optimal ordering policies. To this end, they use simulation techniques to explore the behavior of the supply chain in a wide range of scenarios. The proposed solution takes decisions according to the system state vector, which is generally defined as formed by the inventory position of the various supply chain nodes. In these works, the learning-based approach is shown to outperform different benchmark policies.

A slightly different approach is that by Sui, Gosavi, and Lin (2010) and Akhbari et al. (2014), both focusing on vendor-managed inventory systems. The former employ reinforcement learning for determining the optimal retailer's replenishment policy. Their solution, considering two products, also calculates the number of trucks dispatched by a distribution center to a set of retailers. The latter concentrate on determining the optimal production policy for the manufacturer. They use case-based reasoning by means of the continuous K-nearest neighbor algorithm. Both articles show that the learning-based approach effectively increases the profit of the supply chain over traditional methods.

The usefulness of machine learning for managing the inventory flow through an automatic configuration of the supply chain has also been investigated. Piramuthu (2005a) develops an inductive learning-based tool that determines dynamically the optimal supplier for the different nodes depending on the lead times and the order quantity. Piramuthu (2005b) extends this framework to a multi-product context. In both cases, this dynamic approach, which adjusts the configuration through learning-based techniques, significantly overtakes the one-shoot static configuration in financial terms.

Last, several authors explore the effectiveness of these techniques for demand forecasting, which is an essential part of inventory management. For example, Carbonneau, Laframboise, and Vahidov (2008) show that recurrent neural networks and support vector machines are able to provide very accurate forecasts for real-world datasets, resulting in an improved inventory control. Several recent works follow this research line, see e.g. the reviews by Bajari et al. (2015) and Syam and Sharma (2018).

In line with previous works (e.g. Min 2010, Kuo and Kusiak 2018), we conclude that despite its widespread acceptance as a tool for improving decision-making processes, the applications of machine learning are yet emerging in the supply chain field. There is a wide range of processes that may strongly benefit from the use of these techniques, which would result in strong competitive advantages for firms. It should be highlighted that one of the main advantages of these artificial intelligence techniques is their dynamic nature (Syam and Sharma 2018). This makes them especially suitable for a business scene like the current one, which undergoes rapid and unforeseeable changes.

Our work combines ideas from the above avenues of research but follows a different approach. We contribute to the literature by developing a learning-based framework for setting the most appropriate replenishment policy over time in dynamic environments. Our solution is designed to react to environmental changes; thus considering a wide range of both internal and external factors, as opposed to previous works in this field. Despite the existence of more advanced algorithms, we use inductive learning as it enables a comprehensive decision-making understanding. In this sense, decision trees can be interpreted as "white-box" systems, which allow a deeper analysis of the influencing factors; unlike most machine learning techniques, which are generally considered "black-box" systems (Basse, Charif, and Bódis 2016).

4. Knowledge-based framework for dynamic inventory management

A wide body of literature studies optimal replenishment policies considering their inventory implications in different settings; e.g. see Khouja (1999) for a review in the newsvendor context. The complexity of the problem increases if the production implications of replenishment policies are also considered (Disney et al. 2006); therefore, determining optimal policies in real-world scenarios often becomes an intractable problem. Several methodologies, such as control theory (e.g. Lin, Spiegler, and Naim 2018) and simulation (e.g. Cannella and Ciancimino 2010), have successfully helped to understand the behavior of different policies; however, the question of optimality have been barely addressed. Machine learning techniques can be of special interest in this regard. As previously discussed, they can enable managers to interpret complex interdependences and provide near-optimal solutions to this problem; thus suggesting an interesting avenue for research in the field of inventory management.

In light of this, our approach is built on the dynamic framework for automated inventory management described below[4]. It aims at determining periodically the best inventory model for a node of the supply chain not only according to its state, but also considering the state of its environment. In this sense, this control system is designed to understand the multiple variables, both internal and external, impacting on the node's performance and construct a decision tree that governs the inventory flow. By altering the ordering policy depending on the context where the node operates, we expect to improve the node's operational performance significantly. Figure 2 provides an overview of the inventory management system that we have devised.

Insert Figure 2 about here

Each example includes a combination of values of the relevant variables (input attributes) together with the best inventory policy (class) in this scenario. In the continuous operation of the system, examples for the training set may be obtained from refining the accumulated feedback on its state and performance. However, creating a large mass examples this way may be a very long process. This emphasizes the usefulness of a simulation model that replicates the known environment for populating the example dataset. Through this dataset, the inductive learning algorithm can be capable of acquiring the knowledge and encapsulates it in a decision tree to make future decisions.

The decision tree acts as the regulator of the inventory management system, establishing the replenishment model according to the firm's and the supply chain state over time. Dashed lines in Figure 2 underscore the key role of the supply chain environment in this process, which interacts with the firm in a double way. On the one hand, the supply chain greatly affects the firm's performance —thereby, these factors must also be considered by the control system. On the other hand, the node's decisions impact on its supply chain partners, which creates a hard-to-interpret loop. Considering this external environment and the subsequent emerging interrelationships is a relevant contribution of our framework to the prior literature described in Section 3.

This generic framework can be applied to any kind of supply chain from a single-echelon perspective. No assumptions have been taken on the nature of the supply chain. Nonetheless, according to its conceptual design, this dynamic approach is expected to make a difference in the previously defined fast-changing environments, where the values of the relevant variables rapidly evolve over time. In highly static environments, it may not be necessary to modify (adapt) the inventory policy over time.

Finally, we would like to note that three aspects must be taken into serious consideration in the implementation of the framework. First, system accuracy heavily depends on the attributes; therefore, the key factors must be carefully selected and appropriately measured. Second, achieving a large enough example set is essential to avoid inadequate generalizations that reduce the efficiency of the system. Last, modifying the inventory policy may generate an instable transitory (i.e. changing the policy too frequently may result in poor system performance); therefore, the review period of the dynamic framework must be robustly determined. It is necessary to balance the trade-off between under- and over-reacting to the environmental changes.

5. Simulation model: Generating the training and test examples

5.1. Supply chain scenario and assumptions

To illustrate and evaluate the knowledge-based framework, we consider a node of a supply chain that plays a key role in the distribution of a specific product. This node, labelled as the wholesaler, purchases said product from a factory, which manufactures the product, and later sells it to a retailer, which is the one directly dealing with the consumer. We thus study a single-product serial *supply chain* composed of three nodes, see Figure 3.

Insert Figure 3 about here

The downstream material flow —from the factory to the consumer— comprises three fixed lead times: one production lead time, associated to the manufacturing (T_f), and two shipping lead times, covering the transportation between nodes (T_w, T_r). The upstream information flow —orders travel in the opposite direction— is triggered by the consumer demand. This is considered to follow a normal distribution $N(\mu, \sigma^2)$, where the coefficient of variation *CV*= σ/μ quantifies the uncertainty in the marketplace.

An important assumption behind our supply chain model is that the three nodes operate according to periodic-review inventory policies. Specifically, we adopt the following four-step *sequence of events* (per period, which we understand as a day) for the discrete operation of these nodes, which is common in this kind of studies (e.g. Disney et al. 2016). We do not include the mathematical formulation of the model in full detail due to length restrictions and given that these difference equations are well known in the problem-specific literature.

1. *Reception state*. The product is received (corresponding to the order placed before the relevant lead time) and added to the net stock, and the order is observed. We consider unlimited storage, shipping, and production capacities.
2. *Serving state*. The order received and past backorders (if they exist) are met from net stock. Then, the product is sent downstream. We do not consider defective products, quality loss, or random yields across the supply chain.
3. *Updating state*. The inventory positions (both net stock and work-in-progress) are updated and, if necessary, a backorder is generated. Note that these are allowed, and the product will be delivered as soon as net stock becomes available.
4. *Sourcing state*. The order is issued according to a POUT policy. We assume the quantity cannot be negative, i.e. excess products cannot be returned to the supplier.

POUT models, as per the previous description (in Section 2.2), incorporate four decision points: controller setting, safety stock, forecast, and work-in-progress policy. We consider that the various nodes employ static forecasts $\widehat{D}_t = \mu$, which for normally distributed demands represent minimum mean square error (MMSE) forecasts (Disney et al. 2016). Regarding the work-in-progress policy, we use the common solution $DW_t = T_x\mu$ (where $T_x = \{T_r, T_w, T_f\}$ depending on the node), which allows managers to eliminate a long-term drift in the inventory position (Disney and Towill 2005). Besides, we consider that the safety stock factor is 3, i.e. $SS_t = 3\mu$, in line with prior works in the literature (e.g. Ciancimino et al. 2012). Thus, we focus on the proportional controllers as the main decision variables (retailer: β_r; wholesaler: β_w; factory: β_f).

Finally, we would like to note that this supply chain model has several sources of complexity, e.g. multi-echelon (Ciancimino et al. 2012) and nonlinear effects (Ponte, Wang et al. 2017), which bring it closer to real-world environments but make that determining optimal policies through analytical techniques is an intractable problem. Besides, we would like to underline that we use a generic, instead of specific, supply chain model, as its versatility allows us to draw more comprehensive and generalizable conclusions.

5.2. Example generator and dataset

The example generator is aimed at providing the machine learning algorithm with the necessary information so that it is able to determine the best inventory policy for the wholesaler in each possible scenario. Thus, the class of the examples refer to this optimal policy. In this regard, we model four different policies: (1) *OUT* represents the classic OUT model (i.e. β_w=1); (2) *POUT_H* refers to a POUT model whose controller is regulated at a high level (we select β_w=0.7); (3) *POUT_M* represents a POUT model whose controller is set at a moderate level (we select β_w=0.4); and (4) *POUT_L* refers to a POUT model whose controller is established at a low level (we select β_w=0.1).

In the previously described supply chain scenario, we consider the following attributes to be representative of the node's state and its environment: (1) the coefficient of variation of the demand (*CV*), which ranges between 10% and 50%; (2) the three lead times (T_r, T_w, T_f), which vary between 1 and 4 days, (3) the setting of the retailer's and factory's controller (β_r, β_f), which are randomly generated in the interval [0,1]; and (4) the cost structure of the wholesaler, which is represented by the relative importance of minimizing order variability ($w_o = 1 - w_i$), between 0 and 1 (see Section 2.1).

We implement the simulation model in MATLAB R2014b. The rationale and operation of the example generator are described in Figure 4. After randomly creating the values of the seven attributes, the same scenario is run for the four policies in the wholesaler, which requires previously initializing the system. Each run consists of 20,000 days —a large enough interval to ensure the stability of the response. After the four runs, the class is selected as the policy that obtains the lowest value of the metric J. This generates one example, and the process is repeated until obtaining 2,000 examples. To illustrate this dataset, Table 1 shows an extract.

Insert Figure 4 about here

Insert Table 1 about here

6. Results and discussion

6.1. Accuracy of the inductive learning system

To obtain the inventory management knowledge from the training dataset and structure it through a decision tree, we employ the C4.5 algorithm in the data science software RapidMiner. We use the cross-validation method to validate the results. This randomly divides the example set into ten different blocks, nine of which are employed to obtain the knowledge. The remaining one is used to test the decision tree by calculating the number of examples appropriately classified. We repeat this process ten times and we average the results, which defines the so-called *hit ratio*. This metric reports on the accuracy of the inductive learning algorithm. Figure 5 displays the hit ratio for different sizes of the training dataset (between 100 and 2,000 examples).

Insert Figure 5 about here

As expected, the hit ratio increases as the number of examples grows. Nonetheless, this indicator stabilizes in a narrow range, approx. 87%-89%, over 600 examples. The slight variability would then be mainly explained by the randomness of the examples chosen to validate the algorithm. Overall, we observe that the proposed knowledge-based system is capable of capturing the complex relationships between the different internal and external factors that impact on supply chain performance, determining in approx. 8 out of each 9 scenarios the best replenishment policy for the considered node.

6.2. Decision tree and insights on the impact of the attributes

In this and the next subsection, we consider the knowledge-based control system obtained for 2,000 examples. This contains the most information on the attributes, with the knowledge being structured around 88 decision rules including the seven attributes. By way of illustration, Table 2 reports some of these rules. After each rule, we show the number of examples of the dataset that are properly classified over the total number of examples that verify the conditions of this rule.

Insert Table 2 about here

These 88 rules shape a complex decision tree. For the sake of clarity, we only represent a simplified version of the tree in Figure 6. This shows the branches generated from the two upper variables, respectively, the cost structure of the node represented by w_o and the retailer's inventory controller β_r. At the bottom of this graph, we include the replenishment policies in which each branch ends. Selecting among the different policies in each branch depends on the values of the other attributes.

Insert Figure 6 about here

A major notion derived from the decision tree is the order of relevance of the factors. The tree underscores the weight w_o as the most relevant one. This is interesting but not surprising. It is well known that the optimal value of the inventory controller greatly depends on the cost structure of the node. More unexpected is the finding that the replenishment policy of the retailer (through its controller) is the second factor in terms of importance. This reveals that the ordering policy of the lower echelon of the supply chain greatly impacts on the optimal policy of the wholesaler. Given that the factory's inventory controller is placed much lower in the tree, we interestingly observe that the optimal ordering rule of the wholesaler is more sensitive to the inventory decisions in the lower nodes of the supply chain than to those in the upper nodes. The effect of the different lead times and the demand variability is also less significant than that of the previous attributes.

Moreover, the decision tree allows decision makers to understand the cause-effect relationships between the value of the attributes and their optimal policies. In this regard, Figure 6 shows that when $w_o \leq 0.748$, the inventory controller should never be regulated at low level; while when $w_o > 0.748$, the controller should only be regulated at low or medium level (unless β_r is extremely low). Thus, the more relevant the production costs compared to the inventory costs (i.e. the higher w_o), the stronger the node's motivation to regulate the inventory controller at low levels. Similarly, when β_r is low —and hence the orders issued by the retailer are relatively stable, thus mitigating the Bullwhip Effect—, the wholesaler should opt for high values of the controller. However, when β_r is high —the retailer contributes to amplifying order variability—, the wholesaler should select low values of the controller —which mitigates Bullwhip. For example, if $w_o = 0.8$, the wholesaler should employ an OUT policy or a POUT policy regulated at high level (depending on the other attributes) when $\beta_r = 0.2$, but this node should use a POUT policy with the controller at medium or low level for $\beta_r = 0.8$.

6.3. Comparative analysis against the static system

We now compare the performance of the supply chain operating with the dynamic framework we propose with the static alternative. To this end, we run several simulation runs of 500 months of 30 days. In the static case, the same inventory policy is always employed over time. Meanwhile, in the dynamic framework, we consider that the wholesaler evaluates its internal and external conditions at the beginning of each month, it selects the optimal replenishment policy, and it operates with this policy until next month. That is, the review period of the dynamic framework is set as 30 days.

As previously discussed, the knowledge-based framework has been designed considering the dynamism of the current business scene. From this perspective, we evaluate its performance in two different scenarios. In the first one, labelled as *fast-changing* scenario, the system randomly creates an initial combination of attributes at the beginning of the simulation. Each month, it generates a new combination of attributes by moderately modifying the previous values: within the interval ±1 for the (three) discrete lead times and ±10% for the (four) continuous attributes. In the second one, labelled as *chaotic* scenario, the values of the attributes are randomly generated each month; hence, the attributes may dramatically change from one month to the next one.

Table 3 displays the results of three simulation runs in the fast-changing scenario. In line with previous discussions, we measure operational performance through the average value of the metric J, which is a proxy indicator of the sum of the inventory and production costs incurred by the node. The first four rows show the results of the four policies if they were used statically throughout the whole simulation horizon. The sixth row shows the solution provided by our dynamic approach. For the sake of readability, the values in the table are relative to the lowest possible J (fifth row). This value (1.000), representing the target for each simulation run, would be obtained if the inductive learning algorithm was always capable of selecting the best policy (i.e. hit ratio = 100%).

Insert Table 3 about here

Table 3 provides evidence of how the static approach generates a wide range of avoidable costs in fast-changing environments. The best replenishment policy produces an average J between 18.9% (run 2) and 21.9% (run 3) higher than the target (1.000). These results reveal that the one-shot configuration may be inappropriate in scenarios which undergo significant changes over time. At the same time, Table 3 illustrates that the knowledge-based framework significantly approximates the ideal results. It creates only an increase between 4.5% (run 1) and 6.7% (run 2) in J, thus dramatically outperforming the use of the best policy from a static perspective. In light of this, the dynamic adjustment of the inventory policy in response to the changes in the environmental conditions can significantly contribute to decreasing the wholesaler’s operating costs.

Table 4 presents the results for the chaotic scenario. In this case, the difference between the best static policy and the dynamic solution grows. While the avoidable costs generated by the former increase (the lowest J in the static approach is now around 25% higher than the optimal), those generated by knowledge-based framework are similar as before (the increase in J is slightly above 5%). Note that in this scenario the results vary less between the three runs than before. Similarly, the best static policy here is the POUT_H in the three runs, while in the previous case it was different in each run. This occurs because the results of the fast-changing scenario are much more sensitive to the randomly generated starting point (environmental conditions in each month depend on the previous month, which does not happen in the chaotic scenario).

Insert Table 4 about here

It is important to underline that we have verified statistically that the proposed framework outperforms the best static decision through ANOVA techniques. We have tested the significance of the difference between the means of both alternatives, and we have obtained a p-value much lower than 5%. Thus, we reject the null hypothesis (equality of means) and we confirm the robustness of our findings.

All in all, our results show how real-world businesses may suffer from their inventory strategies becoming obsolete due to the evolving nature of the current business scene. That is, a specific replenishment rule may work well at a certain point in time (i.e. in specific environmental conditions), but it may become inappropriate later on (e.g. if demand uncertainty increases, or if retailers change their inventory policies). From this perspective, we have observed the operational benefits derived from adapting the replenishment rule in response to the changes in the environment, which result in a reduction of Bullwhip-induced costs. Having said this, interpreting the cause-effect relationships between the environmental factors and the optimal policy may become an inextricable problem. In this regard, we demonstrate that the use of machine learning techniques offers an interesting approach to adjusting replenishment rules over time.

7. Conclusions and managerial implications

In today's competitive marketplace, mismanagement of inventories may lead companies to failure. It reduces firm performance by triggering several unnecessary costs, such as those derived from stockouts, holding too much inventory, and unstable production schedules. Determining an appropriate inventory policy then becomes essential. However, in this rapidly changing business scenario, one-shot approaches may not be enough, and companies may benefit from rethinking the suitability of their inventory policy over time.

The present study approaches this problem by proposing a dynamic framework for periodically determining the best replenishment rule for a specific supply chain node. This has been designed to consider both internal and external factors, which constitutes a relevant difference from prior works. Artificial intelligence methods are the backbone of this framework. They can help decision makers to elucidate such a complex problem, which is conditioned by numerous factors whose interactions are hard to interpret.

The first step for practitioners wishing to implement this dynamic approach would be to replicate the known real-world system in a controllable environment, e.g. through a simulation model. This process includes capturing the key variables that impact on operational performance. The model would allow one to explore a wide range of scenarios and investigate the suitability of each inventory policy in them. This information can then be translated into knowledge by a machine learning algorithm, which could establish a set of decision rules for the control of the real-world system over time; thus, equipping firms with decision-making tools to optimize the management of their supply chains.

We have illustrated this process in a simulated case study. An inductive learning algorithm has proven to successfully deal with the convoluted nature of a seven-variable inventory management problem, selecting (among four alternatives) the best inventory policy for a wholesaler with an average accuracy of 88%. This results in a significant reduction of the operating costs in comparison with the best static alternative. The improvement is more accentuated the more rapid and strong changes occur in the business environment. Overall, these outcomes illustrate the high potential of this approach for supply chain practitioners.

We use inductive learning, instead of other machine learning techniques, as it enables the understanding of the decision-making process. In light of this, we have obtained some insights on the impact of the relevant variables on the suitability of the inventory policies. In this regard, the best policy depends primarily on the cost structure of the node. Moreover, our results reveal that the optimal policy is much more sensitive to the inventory policy of the upper echelons than to that of the lower echelons of the supply chain. Interestingly, we have noticed that the optimal policy of the wholesaler depends heavily on whether, or not, the retailer's policy mitigates the Bullwhip Effect.

As future work, we would like to perform a detailed comparative analysis on different machine learning techniques applied to this problem. We plan analyze if the additional complexity that other techniques entail (compared to inductive learning) derive in a noticeable improvement in supply chain performance. The use of model predictive control techniques (Camacho and Bordons 2012) also defines a promising solution strategy for the problem under consideration. Machine learning techniques could also be useful for improving the control of inventories in contexts with inventory inaccuracies, i.e. deviations between the actual and the recorded inventory (e.g. Li and Wang 2018). Another interesting next step could be the exploration of the value of machine learning approaches from the perspective of structural supply chain dynamics through the increasingly popular concept of the *ripple effect* (see Dolgui, Ivanov, and Sokolov 2018). Finally, the adaptation of this framework to closed-loop supply chain archetypes, which incorporate circular economy principles in a bid to reduce environmental impact and leverage economic opportunities (e.g. Goltsos et al. 2018), may also be research directions worth pursuing.

Notes

1. Net stock refers to the end-of-period on-hand inventory. Positive values represent excess inventory (available to satisfy next period's demand), while negative values represent backlogs (unfulfilled demand that still needs to be met); see Disney and Lambrecht (2008).
2. This common cost model considers that a certain guaranteed capacity (GC) is available in each period. If less than GC is needed, labour stands idle for a proportion of the period, hence an opportunity cost is incurred. If more than GC is required, labour works overtime at a higher unit cost, which results in an overtime cost.
3. The absolute gain criterion g_1, representing the information gained by a test T with k outcomes, is defined by $g_1(D,T) = H(D) - \sum_{i=1}^{k} \frac{|D_i|}{|D|} \cdot H(D_i)$; and the relative gain criterion g_2 is defined by $g_2(D,T) = \frac{g_1}{-\sum_{i=1}^{k} \frac{|D_i|}{|D|} \cdot \log_2\left(\frac{|D_i|}{|D|}\right)}$; see Quinlan (1996). In this work, we employ g_2, as g_1 is known to be biased towards tests with many outcomes.
4. The roots of this work are in the models developed by Priore et al. (2001, 2003, 2006, 2010), which use different machine learning techniques for automatically modifying the dispatching rules of flexible manufacturing systems over time. Shiue, Guh, and Lee (2012) review similar approaches in the literature. These works show that this dynamic approach is able to produce breakthrough improvements in performance over the same rules applied statically. This encouraged us to adapt this approach to the supply chain field.

References

Akhbari, M., Y. Z. Mehrjerdi, H. K. Zare, and A. Makui. 2014. "A novel continuous KNN prediction algorithm to improve manufacturing policies in a VMI supply chain." *International Journal of Engineering, Transactions B: Applications* 27 (11): 1681-1690.

APICS. 2011. *APICS Operations Management Body of Knowledge Framework (Third Edition)*. Chicago (IL): APICS The Association for Operations Management.

Axsäter, S. 2003. "Supply chain operations: Serial and distribution inventory systems." *Handbooks in Operations Research and Management Science* 11: 525-559.

Bajari, P., D. Nekipelov, S. P. Ryan, and M. Yang. 2015. "Machine Learning Methods for Demand Estimation." *American Economic Review* 105 (5): 481-85.

Basse, R. M., O. Charif, and K. Bódis. 2016. "Spatial and temporal dimensions of land use change in cross border region of Luxembourg. Development of a hybrid approach integrating GIS, cellular automata and decision learning tree models." *Applied Geography* 67: 94-108.

Bischak, D. P., D. J. Robb, E. A. Silver, and J. D. Blackburn. 2014. "Analysis and Management of Periodic Review, Order- Up- To Level Inventory Systems with Order Crossover." *Production and Operations Management* 23 (5): 762-772.

Camacho, E. F., and C. A. Bordons, C. A. 2012. *Model predictive control in the process industry*. Berlin: Springer Science & Business Media.

Cannella, S., and E. Ciancimino. 2010. "On the bullwhip avoidance phase: supply chain collaboration and order smoothing." *International Journal of Production Research* 48 (22): 6739-6776.

Cannella, S., A. P. Barbosa-Póvoa, J. M. Framinan, and S. Relvas. 2013. "Metrics for bullwhip effect analysis." *Journal of the Operational Research Society* 64 (1): 1-16.

Carbonneau, R., K. Laframboise, and R. Vahidov. 2008. "Application of machine learning techniques for supply chain demand forecasting." *European Journal of Operational Research* 184 (3): 1140-1154.

Cestnik, B., I. Kononenko, and I. Bratko. 1987. "ASSISTANT 86: A knowledge-elicitation tool for sophisticated users." In *Progress in Machine Learning*, edited by I. Bratko and N. Lavrac. Wilmslow (UK): Sigma Press.

Chaharsooghi, S. K., J. Heydari, and S. H. Zegordi. 2008. "A reinforcement learning model for supply chain ordering management: An application to the beer game." *Decision Support Systems* 45: 949-959.

Chopra, S., and M. S. Sodhi. 2004. "Supply-chain breakdown." *MIT Sloan Management Review* 46 (1): 53-61.

Ciancimino, E., S. Cannella, M. Bruccoleri, and J. M. Framinan. 2012. "On the bullwhip avoidance phase: the synchronised supply chain." *European Journal of Operational Research* 221 (1): 49-63.

Dejonckheere, J., S. M. Disney, M. R. Lambrecht, and D. R. Towill. 2003. "Measuring and avoiding the bullwhip effect: A control theoretic approach." *European Journal of Operational Research* 147 (3): 567-590.

Disney, S. M., I. Farasyn, M. Lambrecht, D. R. Towill, and W. Van de Velde. 2006. "Taming the bullwhip effect whilst watching customer service in a single supply chain echelon." *European Journal of Operational Research* 173 (1): 151-172.

Disney, S. M., G. Gaalman, and Hosoda. 2012. "Review of stochastic cost functions for production and inventory control". Paper presented at the 17th International Working Seminar of Production Economics, Innsbruck, 117-128.

Disney, S. M., and M. R. Lambrecht. 2008. "On replenishment rules, forecasting and the bullwhip effect in supply chains." *Foundations and Trends in Technology, Information, and Operations Management* 2 (1): 1–80.

Disney, S. M., A. Maltz, X. Wang, and R. D. Warburton. 2016. "Inventory management for stochastic lead times with order crossovers." *European Journal of Operational Research* 248 (2): 473-486.

Disney, S. M., and D. R. Towill. 2005. "Eliminating drift in inventory and order based production control systems." *International Journal of Production Economics* 93: 331-344.

Dolgui, A., D. Ivanov, and B. Sokolov. 2018. "Ripple effect in the supply chain: an analysis and recent literature." *International Journal of Production Research* 56(1-2), 414-430.

Filipič, B., and M. Junkar. 2000. "Using inductive machine learning to support decision making in machining processes." *Computers in Industry* 43 (1): 31-41.

Friedman, J. H. 1977. "Recursive partitioning decision rule for non-parametric classification." *IEEE Transactions on Computers* 26: 404-408.

Gaalman, G. 2006. "Bullwhip reduction for ARMA demand: The proportional order-up-to policy versus the full-state-feedback policy." *Automatica* 42 (8): 1283-1290.

Giannoccaro, I., and P. Pontrandolfo. 2002. "Inventory management in supply chains: a reinforcement learning approach." *International Journal of Production Economics* 78: 153-161.

Goltsos, T. E., B. Ponte, S. Wang, Y. Liu, M. M. Naim, and A. A. Syntetos. 2018. "The boomerang returns? Accounting for the impact of uncertainties on the dynamics of remanufacturing systems." *International Journal of Production Research*, in press.

Isaksson, O. H., and R. W. Seifert. 2016. "Quantifying the bullwhip effect using two-echelon data: A cross-industry empirical investigation." *International Journal of Production Economics* 171: 311-320.

Kahn, J. A. 1987. "Inventories and the volatility of production." *American Economic Review* 77 (4): 667-679.

Khouja, M. 1999. "The single-period (news-vendor) problem: literature review and suggestions for future research." *Omega* 27 (5): 537-553.

Ko, M., A. Tiwari, and J. Mehnen. 2010. "A review of soft computing applications in supply chain management." *Applied Soft Computing* 10: 661-674.

Kuo, Y. H., and Kusiak, A. 2018. "From data to big data in production research: the past and future trends." *International Journal of Production Research*, in press.

Lancioni, R. A. 2000. "New developments in supply chain management for the millennium." *Industrial Marketing Management* 29 (1): 1-6.

Lee, H. L., V. Padmanabhan, and S. Whang. 1997. "The bullwhip effect in supply chains." *MIT Sloan Management Review* 38 (3): 93.

Li, Q., and S. M. Disney. 2017. "Revisiting rescheduling: MRP nervousness and the bullwhip effect." *International Journal of Production Research* 55 (7): 1992-2012.

Li, M., and Z. Wang. 2018. "An integrated robust replenishment/production/distribution policy under inventory inaccuracy." *International Journal of Production Research* 56 (12): 4115-4131.

Lim, T. S., W. Y. Loh, and Y. S. Shih. 2000. "A comparison of prediction accuracy, complexity, and training time of thirty-three old and new classification algorithms." *Machine Learning* 40 (3): 203-228.

Lin, J., M. M. Naim, L. Purvis, and J. Gosling, J. 2017. "The extension and exploitation of the inventory and order-based production control system archetype from 1982 to 2015." *International Journal of Production Economics* 194: 135-152.

Lin, J., V. L. Spiegler, and M. M. Naim. 2018. "Dynamic analysis and design of a semiconductor supply chain: a control engineering approach." *International Journal of Production Research* 56 (13): 4585-4611.

Mentzer, J. T., W. DeWitt, J. S. Keebler, S. Min, N. W. Nix, C. D. Smith, and Z. G. Zacharia. 2001. "Defining supply chain management." *Journal of Business Logistics* 22 (2): 1-25.

Melnyk, S. A., R. R. Lummus, R. J. Vokurka, L.J. Burns, and J. Sandor. 2009. "Mapping the future of supply chain management: a Delphi study." *International Journal of Production Research* 47 (16): 4629-4653.

Metters, R. 1997. "Quantifying the bullwhip effect in supply chains." *Journal of Operations Management* 15 (2): 89-100.

Michalski, R. S., J. G. Carbonell, and T. M. Mitchell. 1983. *Machine Learning. An Artificial Intelligence Approach*. Paolo Alto (CA): Tioga Publishing Company.

Min, H. 2010. "Artificial intelligence in supply chain management: theory and applications." *International Journal of Logistics Research and Applications* 13 (1): 13-39.

Mortazavi, A., A. A. Khamseh, and P. Azimi. 2015. "Designing of an intelligent self-adaptive model for supply chain ordering management system." *Engineering Applications of Artificial Intelligence* 37: 207-220.

Piramuthu, S. 2005a. "Knowledge-based framework for automated dynamic supply chain configuration." *European Journal of Operational Research* 165 (1): 219-230.

Piramuthu, S. 2005b. "Machine learning for dynamic multi-product supply chain formation." *Expert Systems with Applications* 29 (4): 985-990.

Ponte, B., E. Sierra, D. de la Fuente, and J. Lozano. 2017. "Exploring the interaction of inventory policies across the supply chain: An agent-based approach." *Computers and Operations Research* 78: 335-348.

Ponte, B., X. Wang, D. de la Fuente, and S. M. Disney. 2017. "Exploring nonlinear supply chains: the dynamics of capacity constraints." *International Journal of Production Research* 55 (14): 4053-4067.

Priore, P., D. de la Fuente, A. Gómez, and J. Puente. 2006. "A comparison of machine-learning algorithms for dynamic scheduling of flexible manufacturing systems." *Engineering Applications of Artificial Intelligence* 19: 247-255.

Priore, P., D. de la Fuente, R. Pino, and J. Puente. 2001. "Learning-based scheduling of flexible manufacturing systems using case-based reasoning." *Applied Artificial Intelligence* 15: 949-963.

Priore, P., D. de la Fuente, R. Pino, and J. Puente. 2003. "Dynamic scheduling of flexible manufacturing systems using neural networks and inductive learning." *Integrated Manufacturing Systems* 14 (2): 160-168.

Priore, P., A. Gómez, R. Pino, and R. Rosillo. 2014. "Dynamic scheduling of manufacturing systems using machine learning: an updated review." *Artificial Intelligence for Engineering Design, Analysis and Manufacturing* 28 (1): 83-97.

Priore, P., J. Parreño, R. Pino, A. Gómez, and J. Puente. 2010. "Learning-based scheduling of flexible manufacturing systems using support vector machines." *Applied Artificial Intelligence* 24: 194-209.

Puche, J., B. Ponte, J. Costas, R. Pino, and D. de la Fuente. 2016. "Systemic approach to supply chain management through the viable system model and the theory of constraints." *Production Planning & Control* 27 (5): 421-430.

Quinlan, J. R. 1979. "Discovering rules by induction from large collections of examples". In *Expert Systems in the Micro Electronic Age*, edited by D. Michie. Edinburgh (UK): University Press.

Quinlan, J. R. 1993. *C4.5: Programs for Machine Learning*. San Mateo (CA): Morgan Kaufmann.

Quinlan, J. R. 1996. "Improved use of continuous attributes in C4. 5." *Journal of Artificial Intelligence Research* 4: 77-90.

Rendell, L. A. 1983. "A new basis for state-space learning systems and a successful implementation." *Artificial Intelligence* 20: 369-392.

Sillanpää, V., and J. Liesiö. 2018. "Forecasting replenishment orders in retail: value of modelling low and intermittent consumer demand with distributions." *International Journal of Production Research* 56 (12): 4168-4185.

Steinker, S., M. Pesch, and K. Hoberg. 2016. "Inventory management under financial distress: an empirical analysis." *International Journal of Production Research* 54 (17): 5182-5207.

Shen, B., T. M. Choi, and S. Minner. 2018. "A review on supply chain contracting with information considerations: information updating and information asymmetry." *International Journal of Production Research*, in press.

Shiue, Y. R., R. S. Guh, and K. C. Lee. 2012. "Development of machine learning- based real time scheduling systems: using ensemble based on wrapper feature selection approach." *International Journal of Production Research* 50 (20): 5887-5905.

Sui, Z., A. Gosavi, and L. Lin. 2010. "A reinforcement learning approach for inventory replenishment in vendor-managed inventory systems with consignment inventory." *Engineering Management Journal* 22 (4): 44-53.

Syam, N., and A. Sharma. 2018. "Waiting for a sales renaissance in the fourth industrial revolution: Machine learning and artificial intelligence in sales research and practice.", *Industrial Marketing Management* 69: 135-146.

Watkins, C. I. C. H., and P. Dayan. 1992. "Q-learning." *Machine Learning* 8: 279–292.

Wang, X., and S. M. Disney. 2016. "The bullwhip effect: Progress, trends and directions." *European Journal of Operational Research* 250 (3): 691-701.

Witten, I. H., E. Frank, M. A. Hall, and C. J. Pal. 2016. *Data Mining: Practical machine learning tools and techniques*. Cambridge (MA): Morgan Kaufmann.

Wu, X., V. Kumar, J. R. Quinlan, J. Ghosh, Q. Yang, H. Motoda, G. J. McLachlan, A. Ng, B. Liu, S. Y. Philip, and Z. H. Zhou. 2008. "Top 10 algorithms in data mining." *Knowledge and information systems* 14 (1): 1-37.

Zipkin, P. H. 2000. *Foundations of inventory management*. New York: McGraw-Hill.

Tables and figures

Table 1. Extract of the training set.

	Atributes							*Class*
Example	*A1: T_r*	*A2: T_w*	*A3: T_f*	*A4: CV*	*A5: β_r*	*A6: β_f*	*A7: w_o*	*Policy*
1	1	1	2	16%	0.1328	0.3434	0.1153	OUT
…								
792	2	2	3	30%	0.8447	0.2830	0.4233	POUT_H
793	3	3	3	26%	0.1451	0.0091	0.1556	OUT
794	4	2	3	17%	0.6430	0.5161	0.7269	POUT_M
…								
1466	3	1	2	34%	0.0356	0.2517	0.3727	OUT
1467	2	2	3	42%	0.8034	0.6466	0.8109	POUT_L
1468	2	3	4	23%	0.2413	0.3050	0.7704	POUT_H
…								
2000	2	1	2	17%	0.4290	0.3362	0.9676	POUT_L

Table 2. Extract of the decision rules.

Rule	*If…*	*Then…*	*Hit ratio*
1	$w_o > 0.839$ and $\beta_r > 0.063$ and $\beta_r > 0.086$ and $w_o > 0.842$	POUT_L	263 / 298
2	$w_o > 0.839$ and $\beta_r > 0.063$ and $\beta_r > 0.086$ and $w_o \leq 0.842$ and $\beta_r > 0.624$	POUT_L	4 / 4
3	$w_o > 0.839$ and $\beta_r > 0.063$ and $\beta_r > 0.086$ and $w_o \leq 0.842$ and $\beta_r \leq 0.624$	POUT_M	4 / 4
4	$w_o > 0.839$ and $\beta_r > 0.063$ and $\beta_r \leq 0.086$	OUT	5 / 8
…			
57	$w_o \leq 0.839$ and $w_o \leq 0.748$ and $\beta_r > 0.667$ and $w_o \leq 0.468$ and $w_o > 0.154$ and $\beta_r > 0.671$ and $w_o \leq 0.316$ and $\beta_r \leq 0.746$ and $T_r > 1.500$ and $T_r > 2.500$ and $\beta_f \leq 0.710$ and $\beta_r \leq 0.704$ and $T_f > 2.500$	OUT	1 / 2
58	$w_o \leq 0.839$ and $w_o \leq 0.748$ and $\beta_r > 0.667$ and $w_o \leq 0.468$ and $w_o > 0.154$ and $\beta_r > 0.671$ and $w_o \leq 0.316$ and $\beta_r \leq 0.746$ and $T_r > 1.500$ and $T_r > 2.500$ and $\beta_f \leq 0.710$ and $\beta_r \leq 0.704$ and $T_f \leq 2.500$	POUT_H	2 / 2
…			

Table 3. Results (*J*) in the fast-moving scenario.

Policy	*Run 1*	*Run 2*	*Run 3*	*Mean*
POUT_L (Static)	1.343	*1.189*	1.227	*1.253*
POUT_M (Static)	1.203	1.344	*1.219*	1.256
POUT_H (Static)	*1.201*	1.522	1.295	1.339
OUT (Static)	1.250	1.708	1.402	1.454
MIN (Dynamic)	1.000	1.000	1.000	1.000
INDUCTIVE LEARNING (Dynamic)	1.045	1.067	1.048	1.053
Reduction	(0.156)	(0.123)	(0.172)	(0.200)

Note: We emphasize in italics the best static policy. In parentheses, we show the improvement of the inductive learning-based framework against the best static policy.

Table 4. Results (*J*) in the chaotic scenario.

Policy	*Run 1*	*Run 2*	*Run 3*	*Mean*
POUT_L (Static)	1.478	1.454	1.437	1.456
POUT_M (Static)	1.270	1.277	1.240	1.263
POUT_H (Static)	*1.242*	*1.270*	*1.239*	*1.250*
OUT (Static)	1.268	1.285	1.260	1.271
MIN (Dynamic)	1.000	1.000	1.000	1.000
INDUCTIVE LEARNING (Dynamic)	1.051	1.062	1.051	1.054
Reduction	(0.191)	(0.209)	(0.188)	(0.196)

Note: We emphasize in italics the best static policy. In parentheses, we show the improvement of the inductive learning-based framework against the best static policy.

Figure 1. Problem stages in inductive learning.

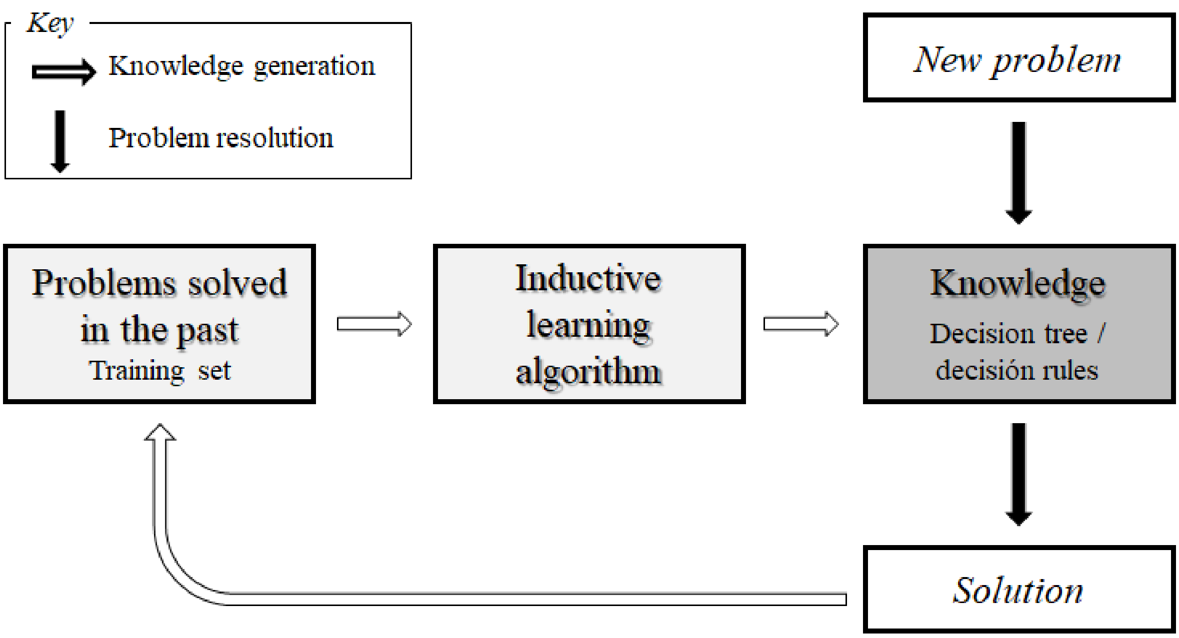

Figure 2. Overview of the knowledge-based framework for automated inventory management

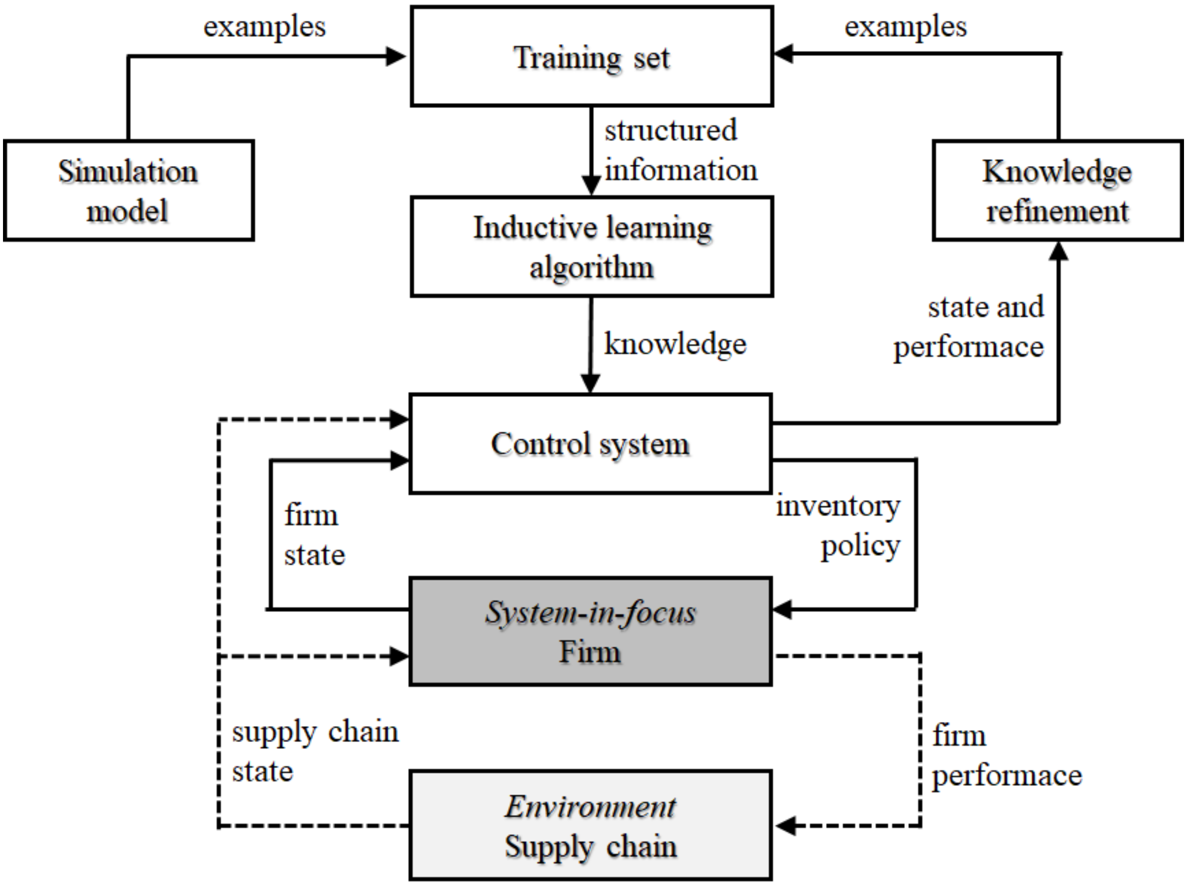

Figure 3. Structure and main variables of the supply chain model.

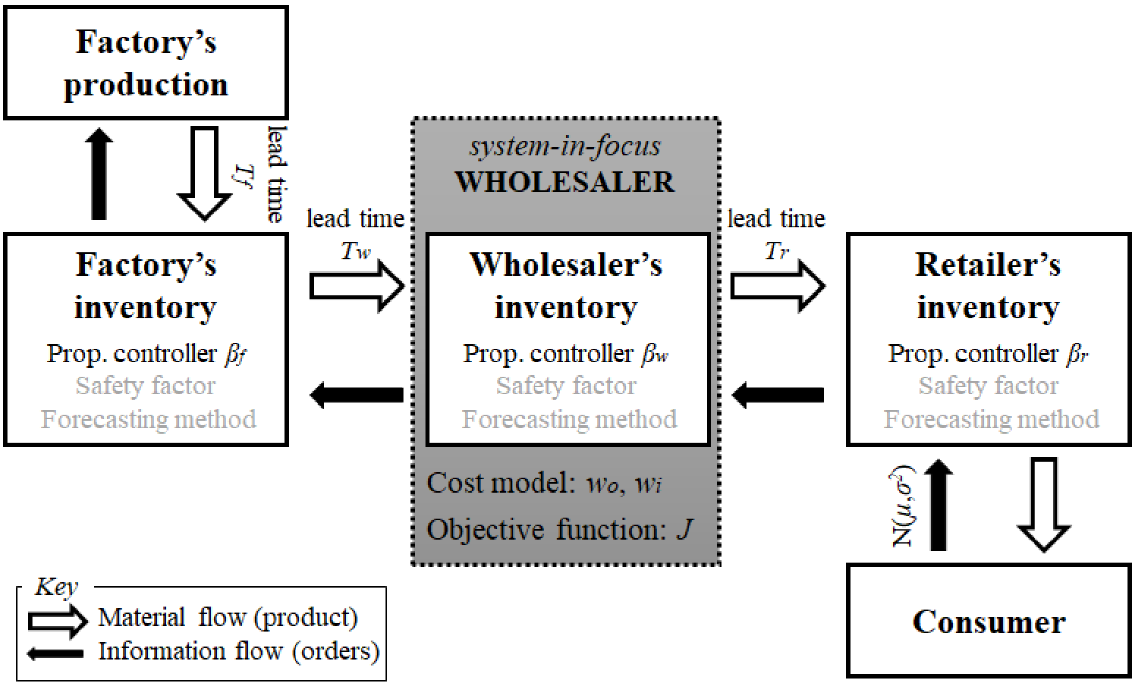

Figure 4. Flow diagram of the example generator.

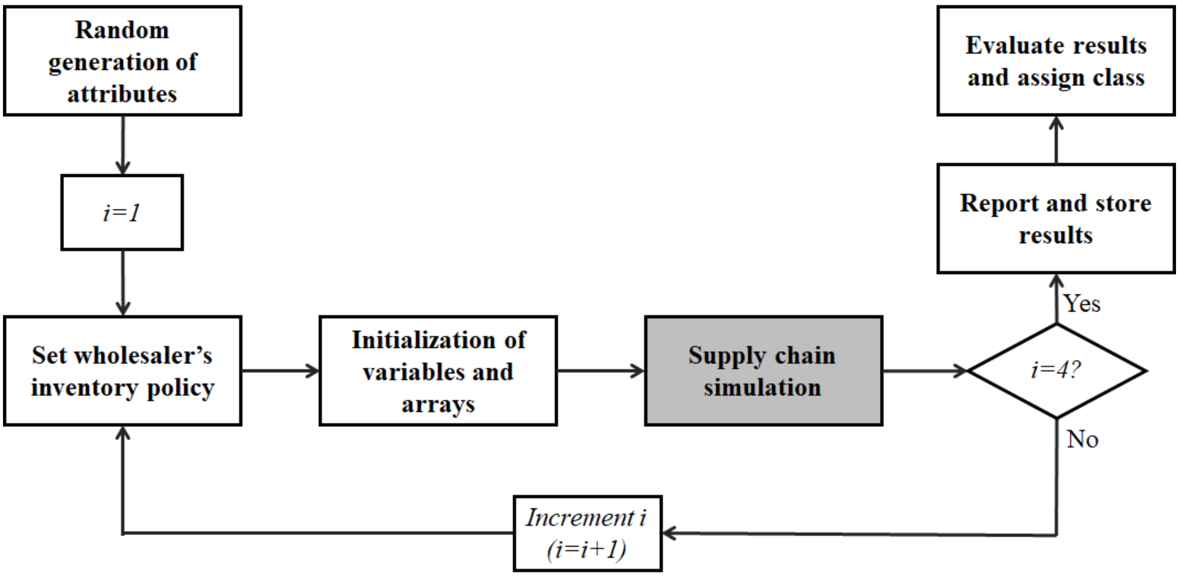

Figure 5. Relationship between the hit ratio and the number of examples in the training set.

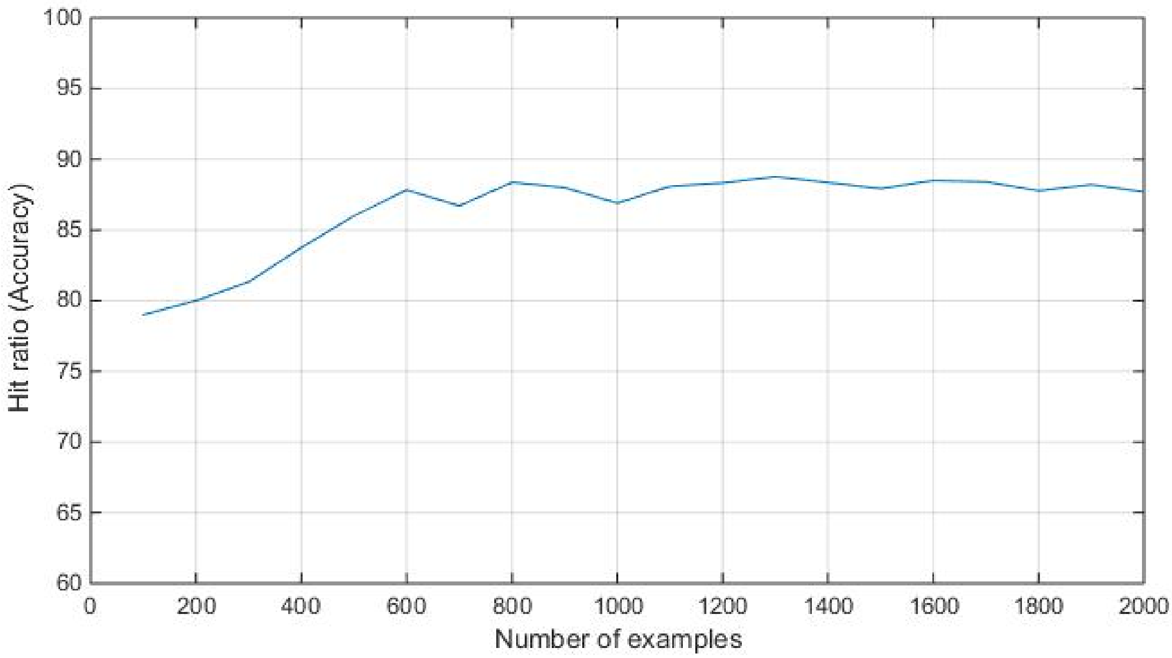

Figure 6. Simplified decision tree generated by the inductive learning algorithm.